International Institute of Information Technology
Bangalore

SUBJECT VLSI ARCHITECTURE DESIGN

COURSECODE VL 731

Project

Yathin Kumar
(IMT2020550)

B Sathiya Naraayanan
(IMT2020534)

Maurya Patel
(IMT2020517)

Paras Vekariya
(IMT2020547)

March 14, 2023

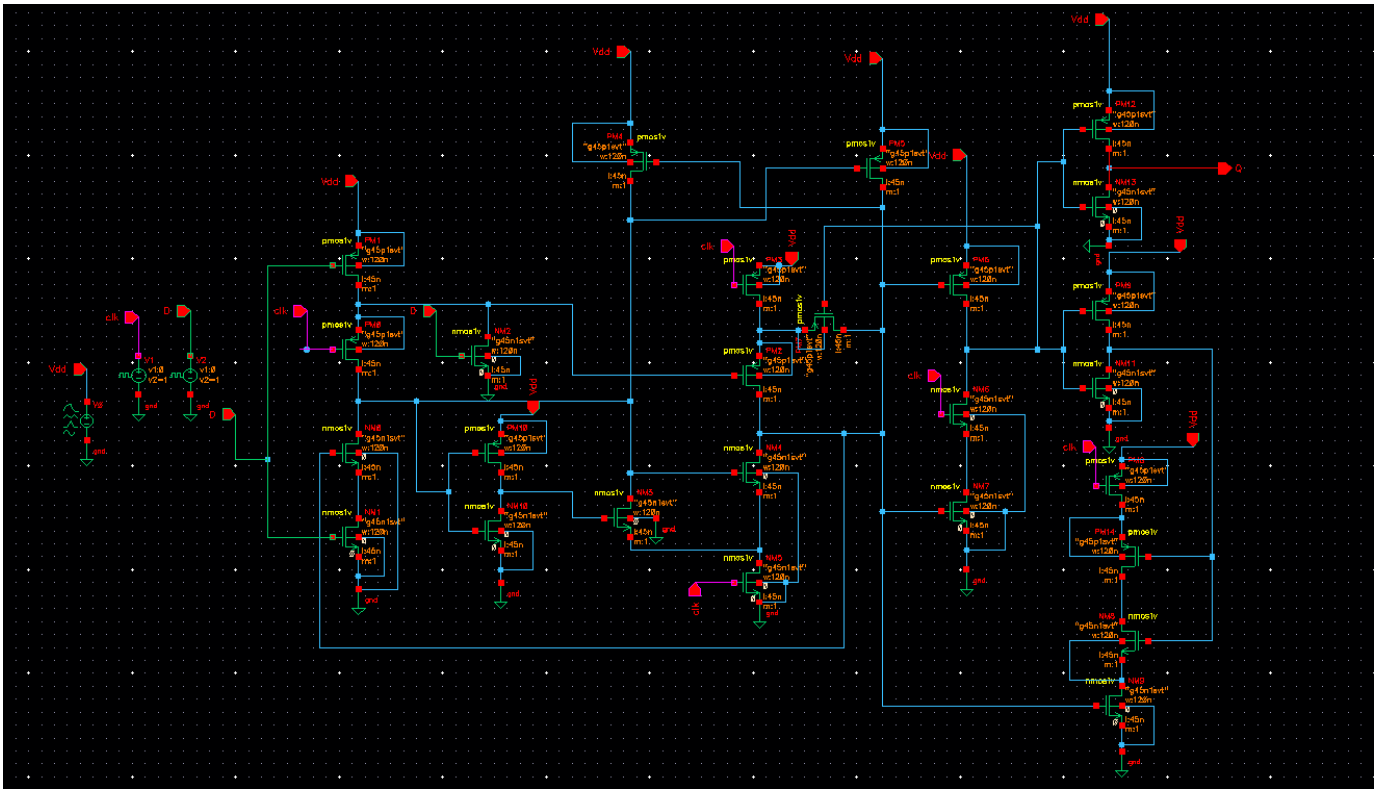

iit-b
ज्ञानमुत्तमम्

1 PROPOSED ARCHITECTURE

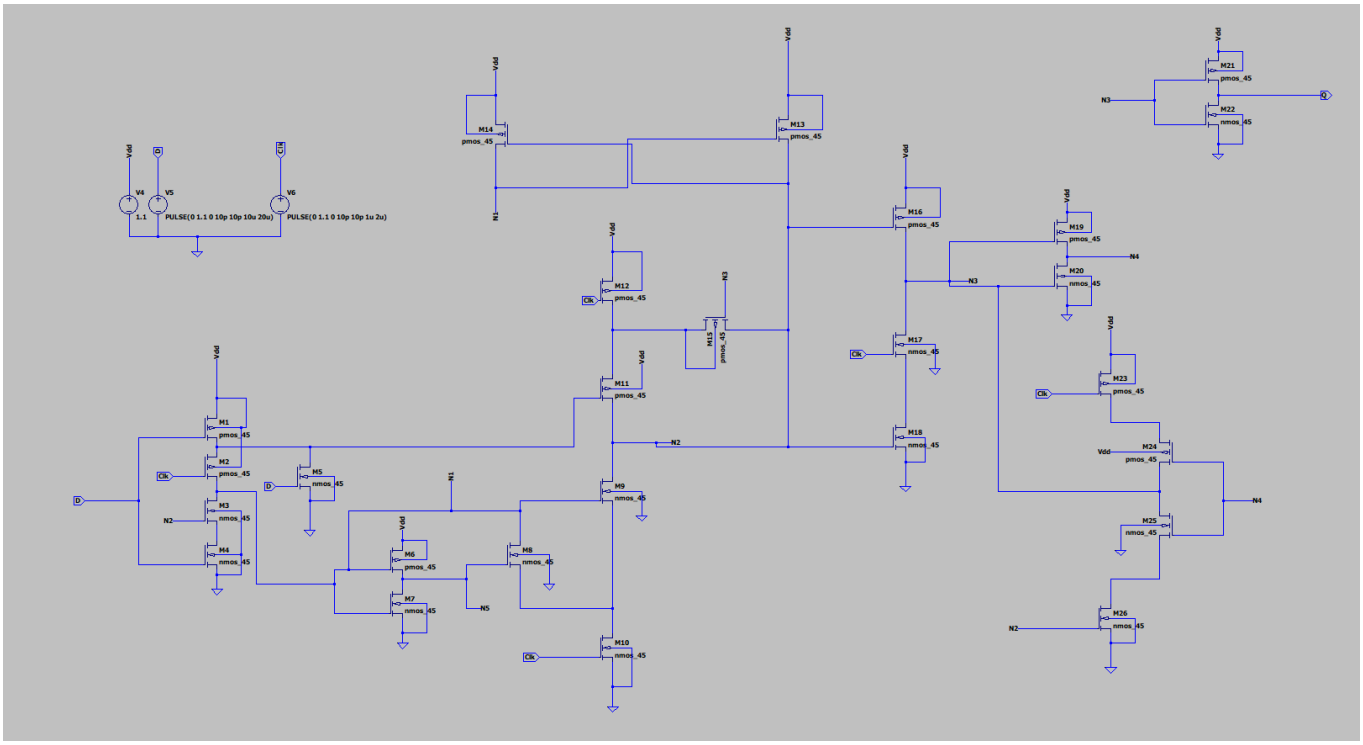
We are simulating the ouput Q by giving a clock and D as input from LtSpice for the proposed architecture.With the employment of input-aware precharge scheme, the proposed TSPC FF precharges only when necessary

1.1 SCHEMATIC

1.1.1 CADENCE VIRTUOSO

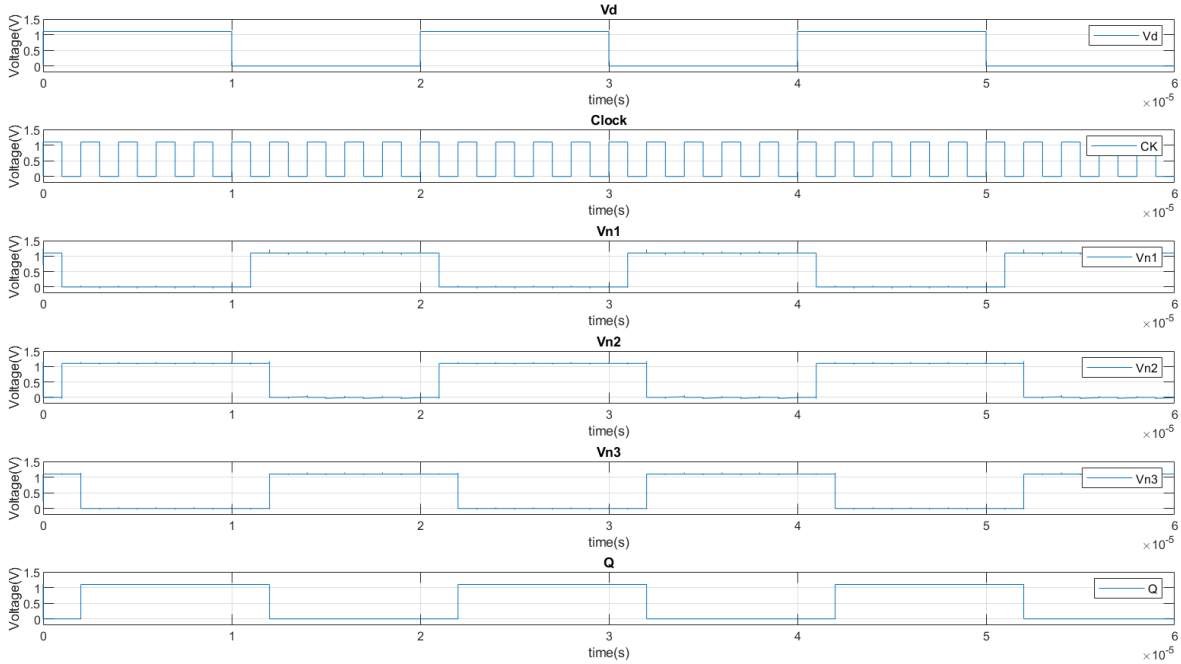


1.1.2 LTSPICE

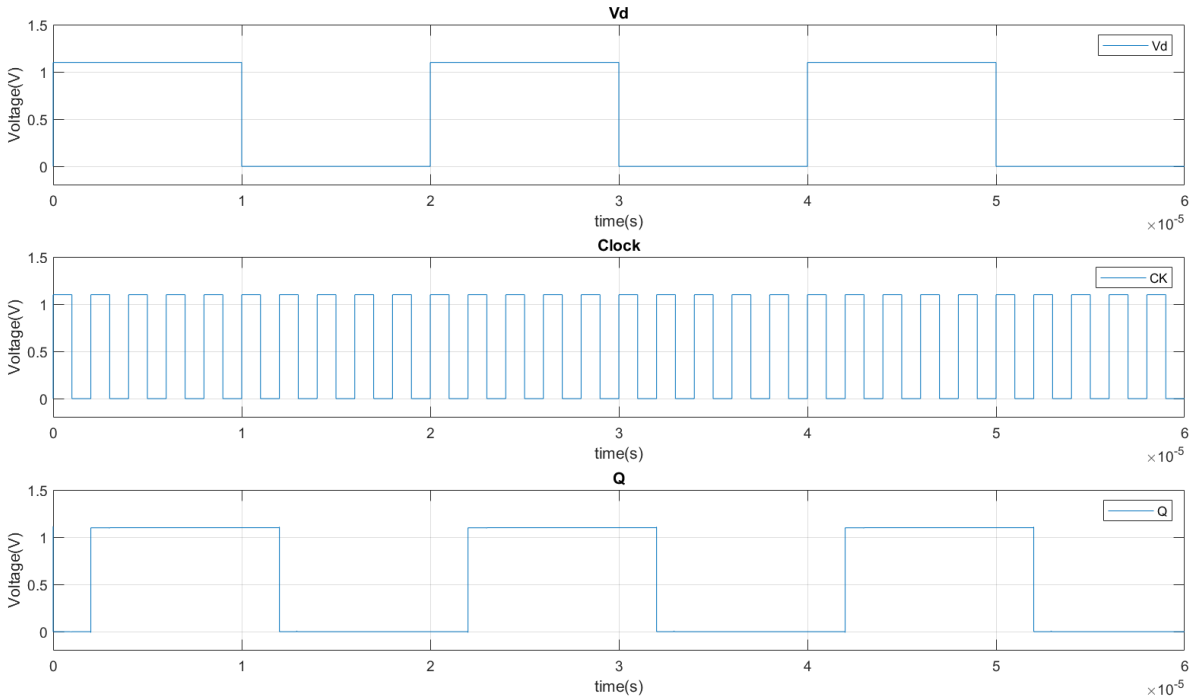


1.2 PLOTS

1.2.1



1.2.2

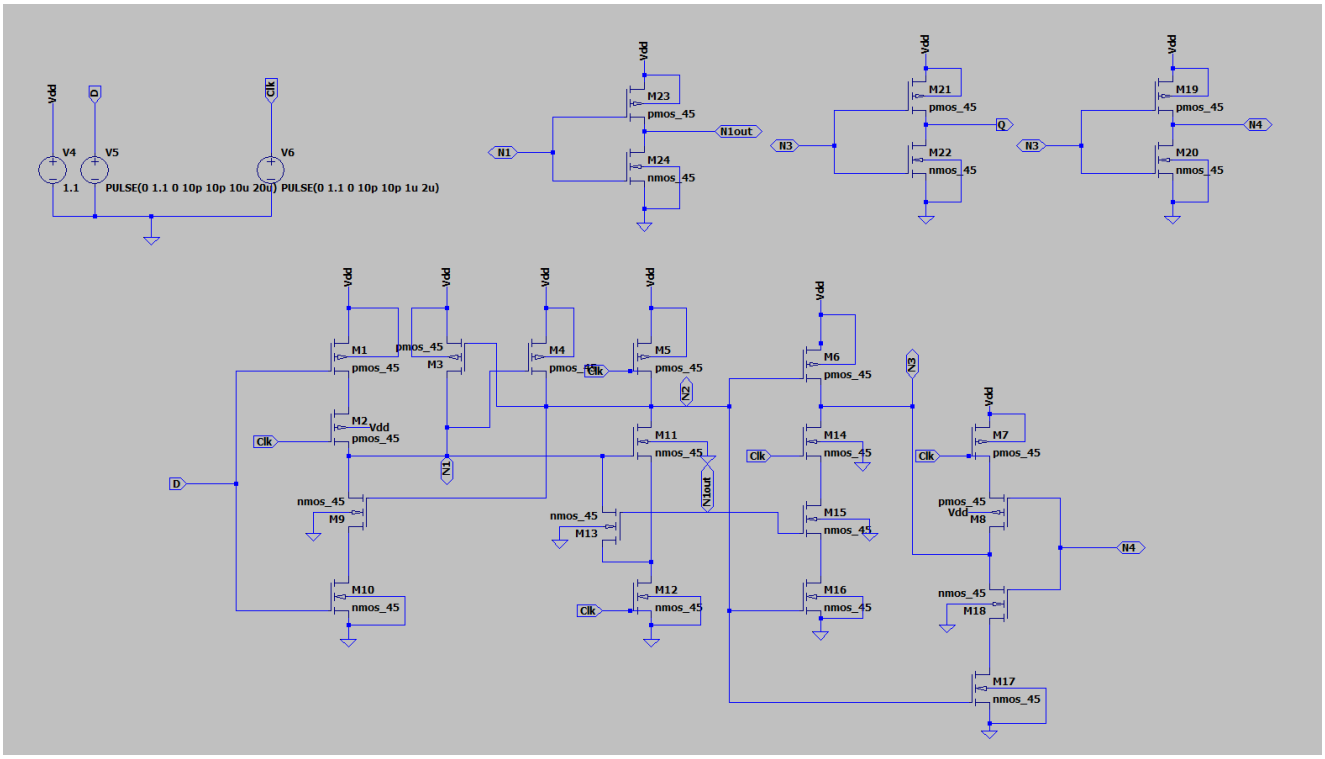

Vd
Voltage(V)
time(s)
×10^-5
Vd
Clock
Voltage(V)
time(s)
×10^-5
CK
Q
Voltage(V)
time(s)
×10^-5
Q

2 SSCFF

We are simulating the ouput Q by giving a clock and D as input from LtSpice for the SSCFF architecture.It is Static Single-Phase Contention-free flip flop.The main drawback is redundant precharges because of lack of input-aware precharge.

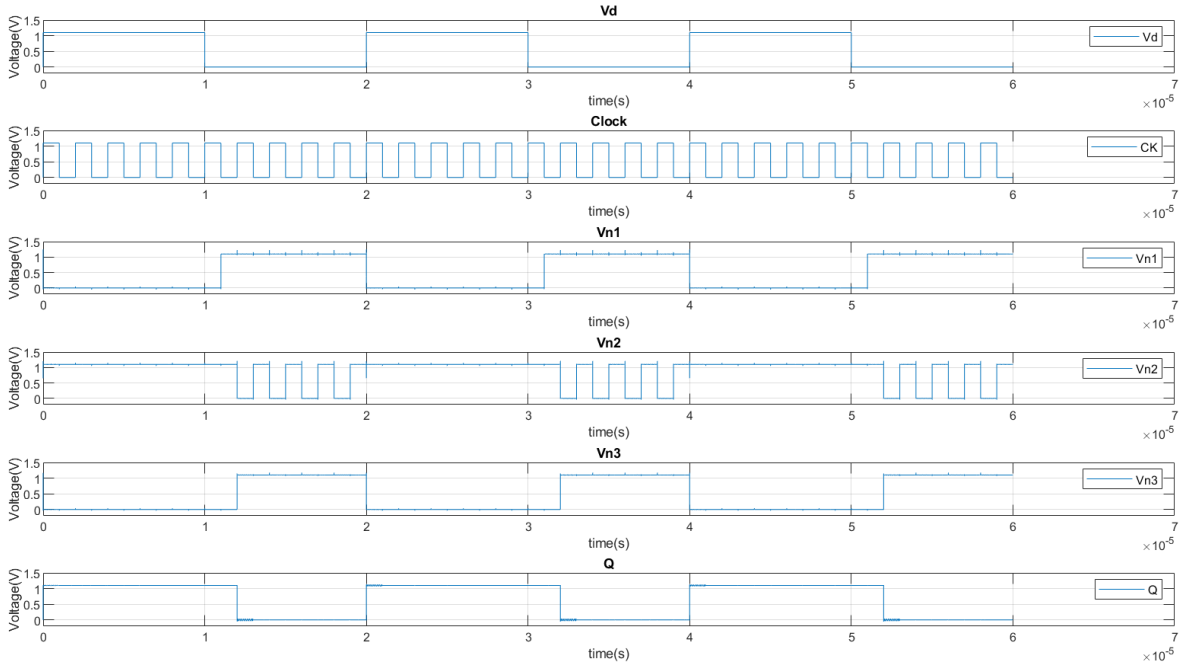
2.1 SCHEMATIC

2.1.1 LTSPICE

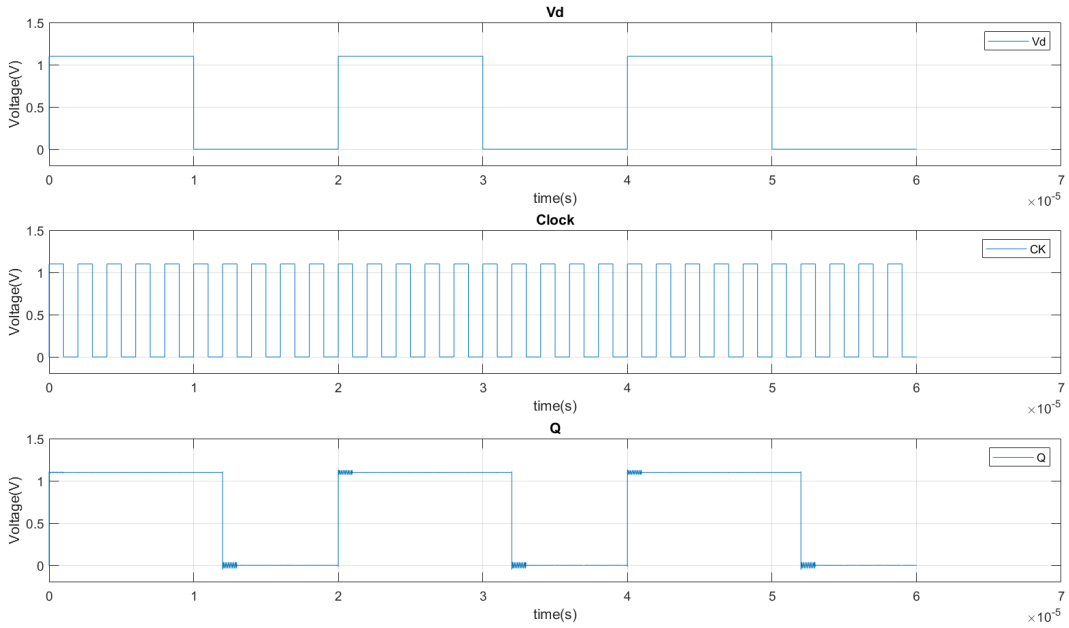


2.2 PLOTS

2.2.1



2.3

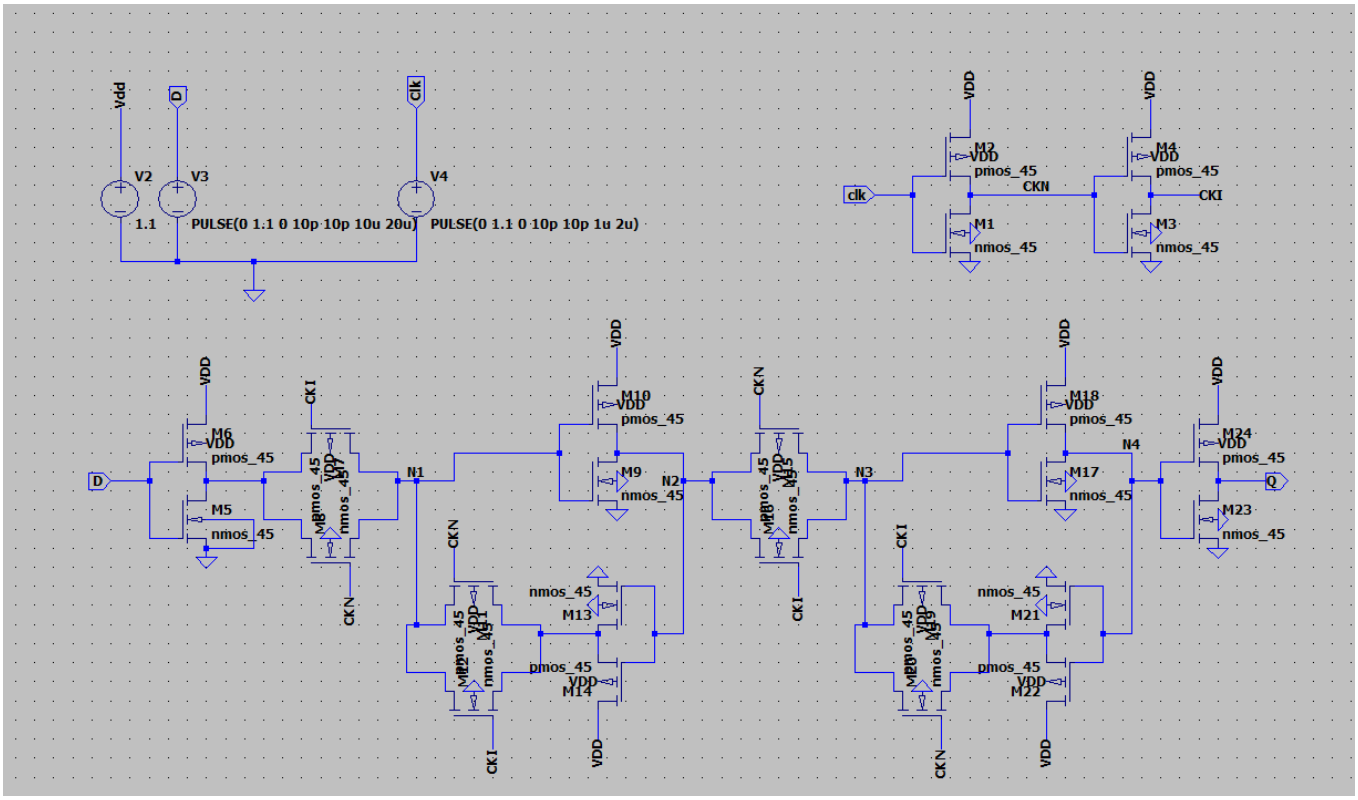


3 TGFF

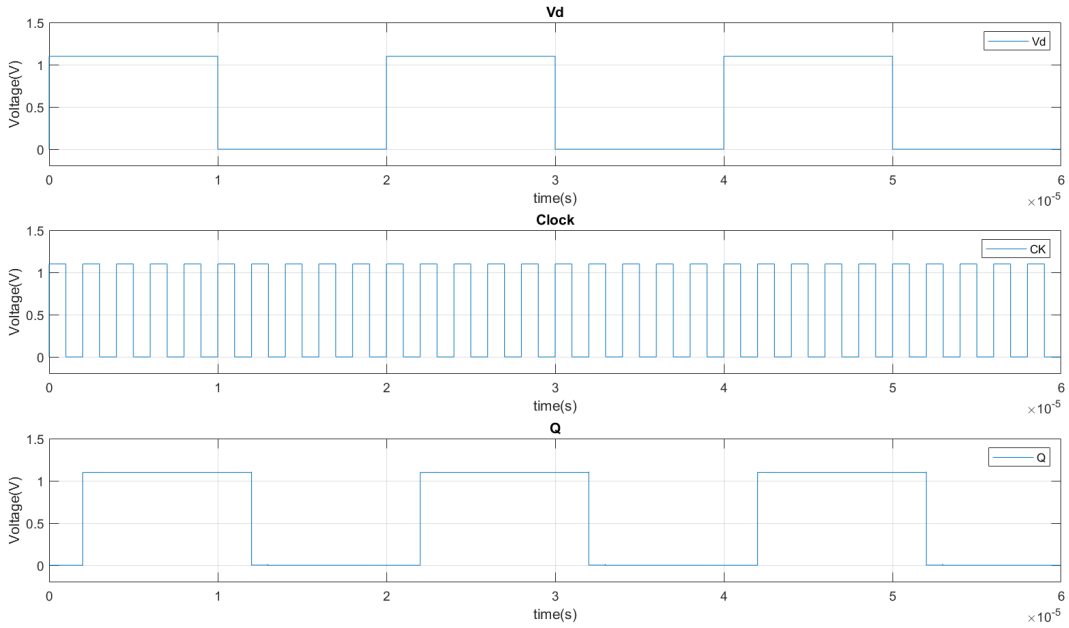
The TGFF is a contention-free FF which is suitable for near-threshold operation. The main drawback of TGFF is the large clock network.The internal nodes CKN and CKI toggle no matter what the input data is, and the nodes CKN and CKI drive a larger number of transistors. Thus,the power consumption of TGFF is still large even if the data activity remains low.

3.1 SCHEMATIC

3.1.1 LTSPICE



3.2 PLOTS


Vd
Voltage(V)
time(s)
×10^-5
Vd
Clock
Voltage(V)
time(s)
×10^-5
CK
Q
Voltage(V)
time(s)
×10^-5
Q

4 ANALYSIS

4.1 THEORY

The analysis was done with ptm low power 45nm technology in LtSpice.The dealy from clock to Q was calculated by subtracting the time taken by Q to reach 50% of its maximum value by time taken by CK for the same.Average power was calculated by multiplying supply voltage and source current.The plots are for 10% data activity.The technology node for TGFF was not mentioned in the paper,so 45 nm was used for analysis.We used PSO algorithm for finding the optimal widths.

PSO Code used:

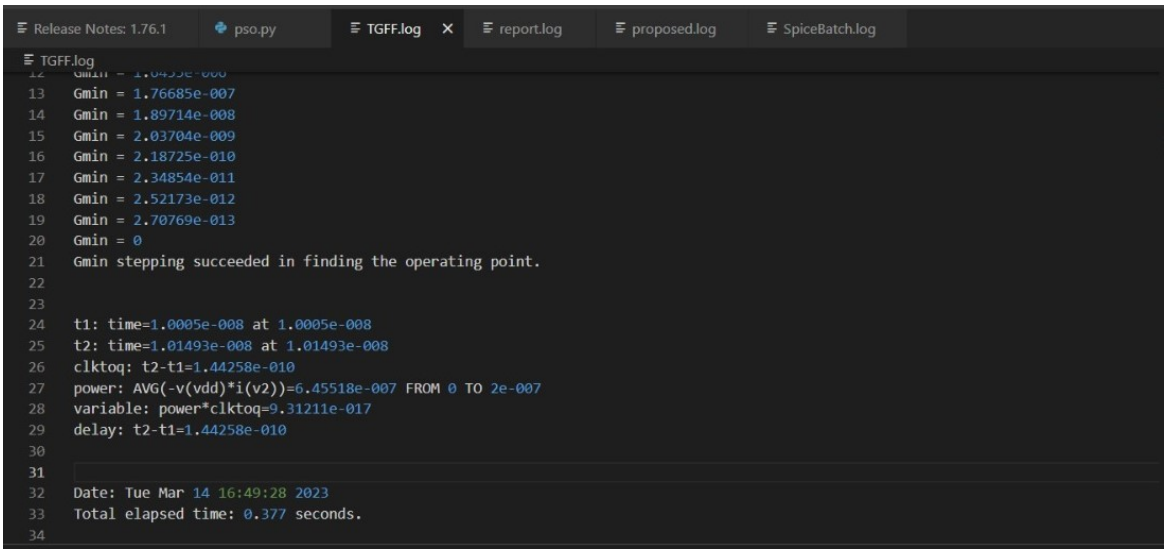
```
import os
from PyLTSpice.LTSpiceBatch import SimCommander
from PyLTSpice.LTSteps import LTSpiceLogReader
import numpy as np
import pyswarms as ps
from pyswarms.utils.functions import single_obj as fx


def LTSpice_PSO(W):
    pdp = [100.0]*10
    for i in range(10):
        LTC = SimCommander("proposed.asc")
        LTC.set_parameters(w1=W[i][0], w2=W[i][1], w3=W[i][2], w4=W[i][3], w5=W[i]
        [4], w6=W[i][5], w7=W[i][6], w8=W[i][7], w9=W[i][8], w10=W[i][9])
        LTC.run()
        LTC.wait_completion()
        f = open("proposed.log", "r")
        data = f.read()
        x = data.split("\n")
        for j in range(len(x)):
            if x[j][0:6] == "delay:":
                print(x[j])
                pdp_str = x[j].split("=")
                pdp[i] = float(pdp_str[1])
                break
    return pdp


min_bound = np.ones(10)*90e-9
max_bound = np.ones(10)*240e-9
bounds = (min_bound, max_bound)
options = {'c1': 0.5, 'c2': 0.3, 'w':0.9}
optimizer = ps.single.GlobalBestPSO(n_particles=10, dimensions=10,options=options,
bounds=bounds)
cost, pos = optimizer.optimize(LTSpice_PSO, iters=30)
```

Results in the paper:
Delay (Clock - Q) - TGFF:104.73 ps; S2CFF:73.11 ps; Proposed:77.31 ps
Normalized Power - TGFF:1037.93 nW; S2CFF:360.79 nW; Proposed:162.18 nW

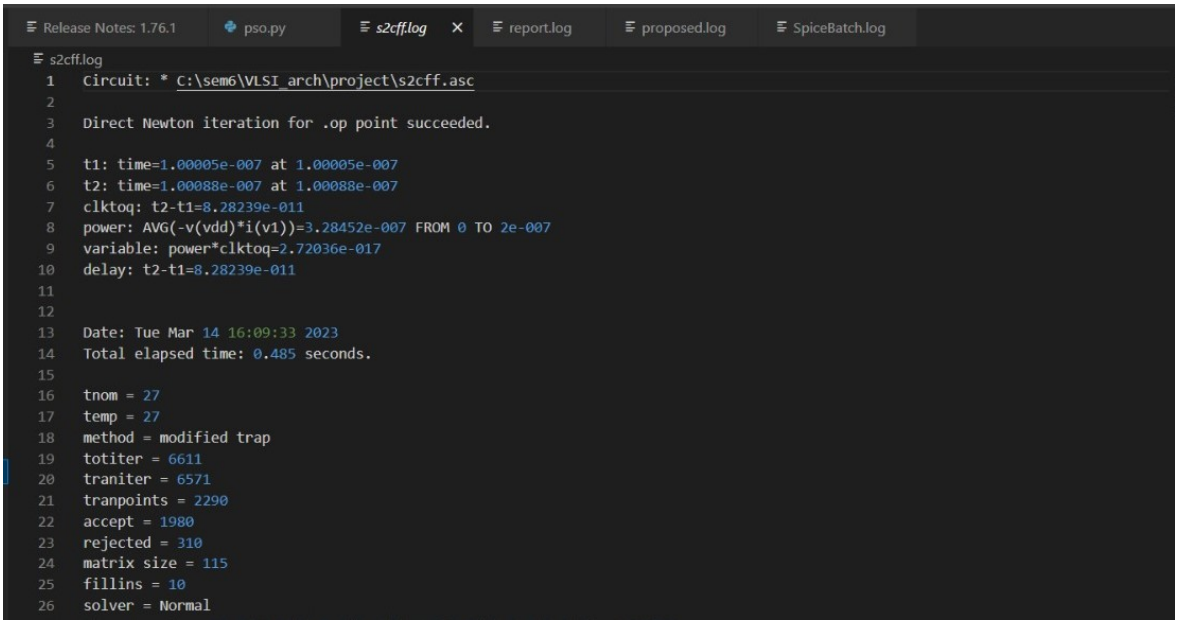
Obtained Results:
Delay (Clock - Q) - TGFF:144 ps; S2CFF:82.8 ps; Proposed:24.4 ps
Normalized Power - TGFF:645 nW; S2CFF:328 nW; Proposed:220 nW

4.2 PLOTS

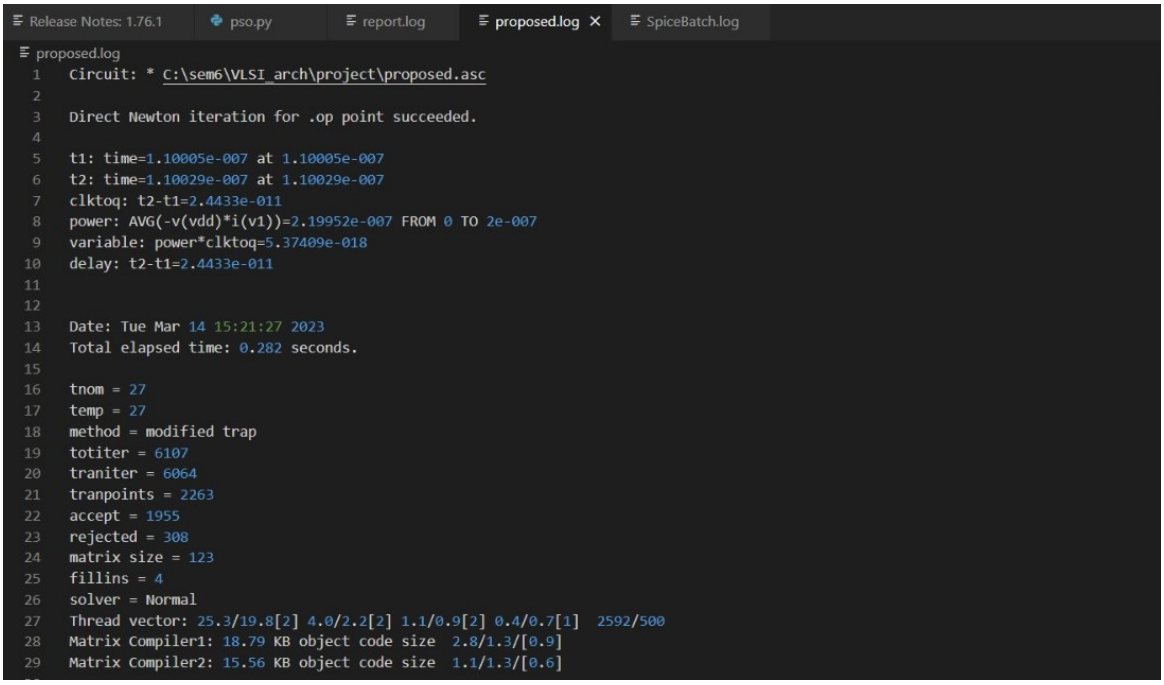
4.2.1



4.2.2



4.2.3



5 FUTURE WORK

We are simulating the mentioned architectures in cadence virtuoso and will obtain setup and holdtime.It will be compared for different data activity.We will plot Schematic of the proposed FF with (a) set, (b) reset, and (c) scan.